

UML – Unified Modeling Language

Die UML ist eine standardisierte grafische Sprache, die in der Informatik verwendet wird, um Software-Systeme zu modellieren. Es gibt mehrere verschiedene Diagrammarten in UML, von denen jede eine spezifische Funktion hat. Hier sind die wichtigsten:

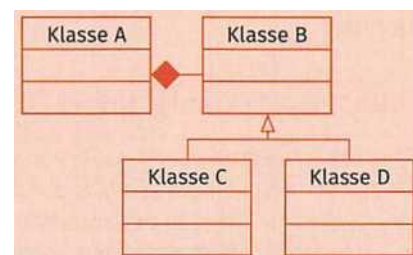
1. Klassendiagramme: Sie zeigen die statische Struktur eines Systems durch die Darstellung von Klassen, Attributen, Methoden und Beziehungen zwischen ihnen.
2. Anwendungsfalldiagramme: Sie beschreiben die funktionalen Anforderungen an ein System aus der Sicht des Benutzers und zeigen, welche Aktionen der Benutzer ausführen kann und welche Reaktionen das System darauf liefert.
3. Sequenzdiagramme: Sie zeigen die Interaktionen zwischen den Objekten im System und die Reihenfolge, in der sie auftreten.
4. Zustandsdiagramme: Sie beschreiben den Lebenszyklus eines Objekts und zeigen die verschiedenen Zustände, in denen sich das Objekt während seiner Lebenszeit befinden kann.
5. Aktivitätsdiagramme: Sie zeigen den Ablauf von Aktivitäten oder Prozessen in einem System und beschreiben, wie Objekte und Ressourcen im System miteinander interagieren.
6. Komponentendiagramme: Sie zeigen die physische Struktur eines Systems und die Abhängigkeiten zwischen den Komponenten.
7. Verteilungsdiagramme: Sie zeigen die physische Verteilung von Komponenten und Systemen in einer Netzwerkumgebung.

Diese Diagrammarten bieten einen breiten Überblick über die verschiedenen Aspekte eines Systems und sind ein wichtiges Werkzeug bei der Planung, Entwicklung und Dokumentation von Software-Projekten.

Klassendiagramm

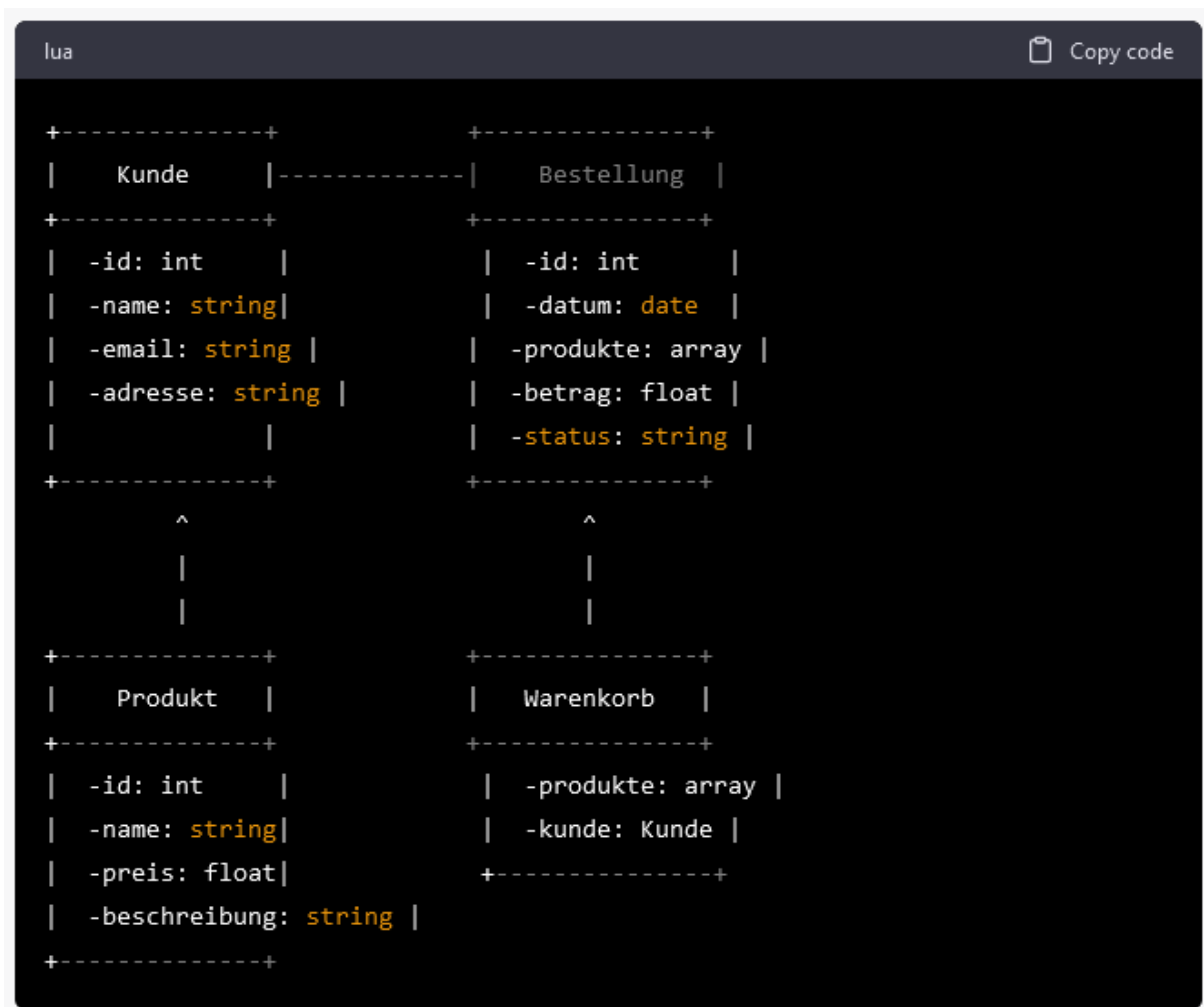
Das Klassendiagramm ist eine Diagrammart in UML, die die statische Struktur eines Systems modelliert. Es beschreibt die Klassen und deren Beziehungen zueinander sowie die Attribute und Methoden, die jede Klasse besitzt.

Ein einfaches Beispiel für ein Klassendiagramm könnte ein Online-Shop-System sein. In diesem System gibt es verschiedene Klassen, wie zum Beispiel "Kunde", "Produkt", "Bestellung", "Warenkorb", "Zahlung" usw.



Die Klassen werden als Rechtecke mit dem Namen der Klasse im Inneren dargestellt. Innerhalb des Rechtecks werden auch die Attribute und Methoden der Klasse aufgeführt. Die Beziehungen zwischen den Klassen werden durch Linien dargestellt, die die Verbindung zwischen den Klassen anzeigen.

Ein Beispiel für ein solches Klassendiagramm könnte wie folgt aussehen:



In diesem Beispiel hat der "Kunde" eine Beziehung zur "Bestellung" und zum "Warenkorb", während die "Bestellung" eine Beziehung zum "Produkt" hat. Die Attribute und Methoden jeder Klasse sind ebenfalls aufgeführt.

Verbindungen in Klassendiagrammen

In der UML gibt es mehrere Verbindungen, die verwendet werden können, um die Beziehungen zwischen Klassen in einem Klassendiagramm darzustellen. Hier sind einige der wichtigsten Verbindungen:

1. Assoziation: Eine Assoziation stellt eine Verbindung zwischen zwei Klassen her, um anzuzeigen, dass sie in Beziehung zueinander stehen. Eine Assoziation kann unidirektional oder bidirektional sein. Die Assoziation kann eine Eigenschaft haben, die angibt, wie viele Objekte jeder Klasse beteiligt sind und welche Rolle sie spielen.

Beispiel: Ein Beispiel für eine Assoziation ist die Beziehung zwischen einem Lehrer und einem Schüler. Jeder Lehrer kann mehrere Schüler haben, und jeder Schüler hat genau einen Lehrer. Diese Beziehung kann mit einer unidirektionalen Assoziation von der Klasse "Lehrer" zur Klasse "Schüler" dargestellt werden.

2. Aggregation: Die Aggregation ist eine spezielle Art von Assoziation, die eine Teil-Ganzes-Beziehung zwischen Klassen darstellt. Hierbei wird eine Klasse als Ganzes

und eine andere Klasse als Teil dieser Klasse dargestellt. Ein Teil kann jedoch auch zu mehreren Ganzen gehören.

Beispiel: Ein Beispiel für eine Aggregation ist die Beziehung zwischen einer Universität und ihren Fakultäten. Eine Fakultät ist ein Teil der Universität, aber sie kann auch zu anderen Universitäten gehören. Diese Beziehung kann mit einer Aggregationsbeziehung von der Klasse "Universität" zur Klasse "Fakultät" dargestellt werden.

3. **Komposition:** Die Komposition ist eine weitere spezielle Art von Assoziation, die eine stärkere Teil-Ganzes-Beziehung zwischen Klassen darstellt. Hierbei wird eine Klasse als Ganzes und eine andere Klasse als Teil dieser Klasse dargestellt. Der Teil kann jedoch nur zu einem Ganzen gehören und existiert nicht mehr, wenn das Ganze zerstört wird.

Beispiel: Ein Beispiel für eine Komposition ist die Beziehung zwischen einem Auto und seinen Reifen. Die Reifen sind ein Teil des Autos, aber sie können nicht zu anderen Autos gehören. Wenn das Auto zerstört wird, existieren die Reifen auch nicht mehr. Diese Beziehung kann mit einer Kompositionsbeziehung von der Klasse "Auto" zur Klasse "Reifen" dargestellt werden.

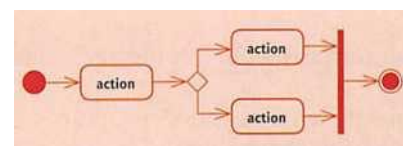
4. **Vererbung:** Die Vererbung ist eine spezielle Beziehung zwischen Klassen, bei der eine Unterklasse eine spezialisierte Version der Oberklasse darstellt. Die Unterklasse erbt die Eigenschaften und Methoden der Oberklasse und kann diese überschreiben oder erweitern.

Beispiel: Ein Beispiel für eine Vererbungsbeziehung ist die Beziehung zwischen einer Tierklasse und ihren Unterarten wie Hund, Katze und Pferd. Diese Unterarten haben alle Eigenschaften und Methoden der Tierklasse geerbt, aber sie können auch eigene spezialisierte Eigenschaften und Methoden haben.

Diese Verbindungen ermöglichen es, komplexe Beziehungen zwischen Klassen in einem Klassendiagramm darzustellen und somit die Struktur eines Systems zu modellieren.

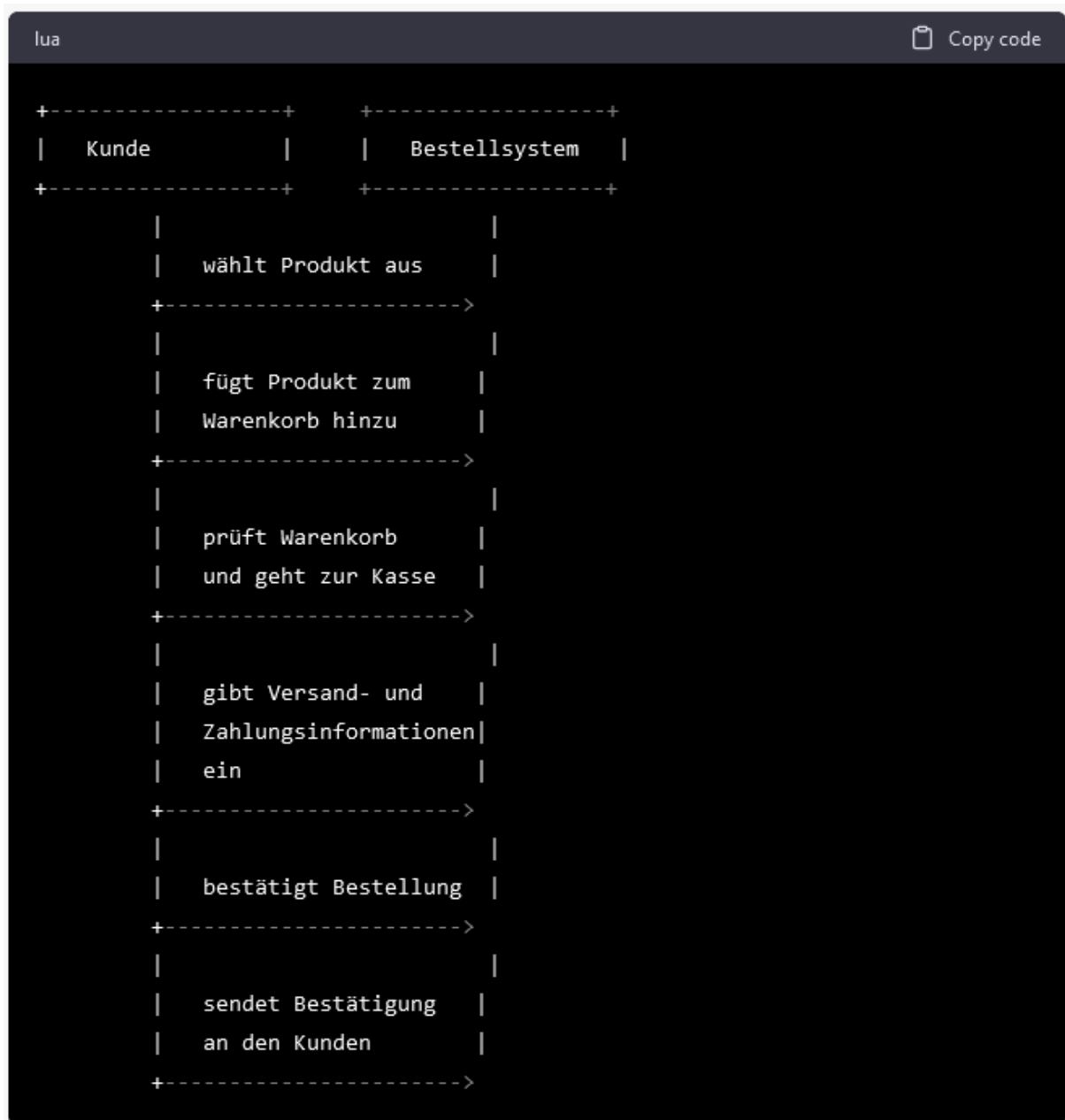
Aktivitätsdiagramm

Das Aktivitätsdiagramm ist eine Diagrammart in UML, die den Ablauf von Aktivitäten oder Prozessen in einem System modelliert. Es beschreibt, wie Objekte und Ressourcen im System miteinander interagieren und welche Schritte ausgeführt werden, um ein bestimmtes Ergebnis zu erzielen.



Ein einfaches Beispiel für ein Aktivitätsdiagramm könnte ein Bestellprozess in einem Online-Shop-System sein. In diesem Prozess gibt es verschiedene Schritte, die ausgeführt werden müssen, um eine Bestellung aufzugeben.

Das Aktivitätsdiagramm könnte wie folgt aussehen:

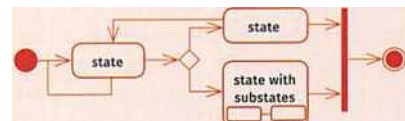


In diesem Beispiel gibt es zwei beteiligte Parteien: den "Kunden" und das "Bestellsystem". Der Kunde wählt zunächst ein Produkt aus und fügt es zum Warenkorb hinzu. Dann prüft er den Warenkorb und geht zur Kasse, wo er Versand- und Zahlungsinformationen eingibt. Schließlich bestätigt er die Bestellung und das System sendet ihm eine Bestätigung per E-Mail.

Das Aktivitätsdiagramm gibt einen Überblick über den gesamten Prozess und zeigt die Interaktionen zwischen dem Kunden und dem System auf. Es ist ein wichtiges Werkzeug bei der Planung und Entwicklung von komplexen Abläufen und Prozessen in einem System.

Zustandsdiagramm

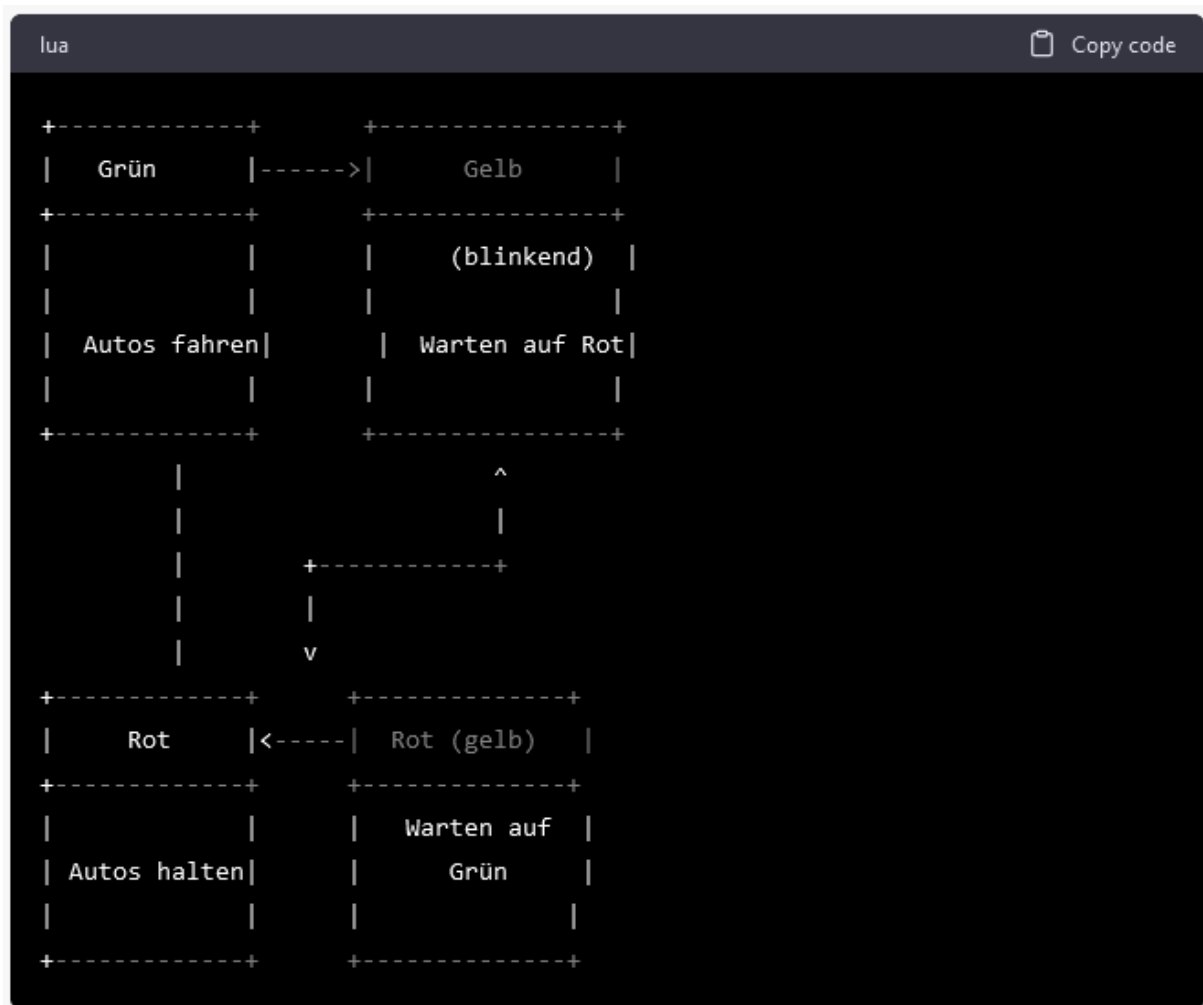
Das Zustandsdiagramm ist eine Diagrammart in UML, die den Lebenszyklus von Objekten in einem System modelliert.



Es beschreibt die verschiedenen Zustände, die ein Objekt durchlaufen kann, und die Übergänge zwischen diesen Zuständen.

Ein einfaches Beispiel für ein Zustandsdiagramm könnte ein Ampelsystem sein. In diesem System gibt es verschiedene Zustände, die von der Ampel eingenommen werden können, wie zum Beispiel "Grün", "Gelb" und "Rot".

Das Zustandsdiagramm könnte wie folgt aussehen:

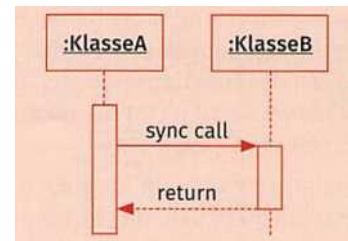


In diesem Beispiel gibt es fünf Zustände, die von der Ampel durchlaufen werden können. Wenn die Ampel grün ist, können Autos fahren. Wenn die Ampel gelb wird, bedeutet das, dass die Ampel in Kürze auf Rot wechselt. Wenn die Ampel rot wird, müssen Autos anhalten, bis sie wieder grün wird. Wenn die Ampel Rot (gelb) wird, bedeutet das, dass die Ampel gleich grün wird. Wenn die Ampel Grün ist und der Verkehr fließt, können Autos fahren.

Das Zustandsdiagramm gibt einen Überblick über den Lebenszyklus der Ampel und zeigt, welche Übergänge zwischen den Zuständen möglich sind. Es ist ein wichtiges Werkzeug bei der Modellierung von Systemen, die Zustände durchlaufen, wie zum Beispiel Automaten oder Prozessoren.

Sequenzdiagramm

Das Sequenzdiagramm ist eine Diagrammart in UML, die die Interaktionen zwischen verschiedenen Objekten in einem System darstellt. Es zeigt die zeitliche Abfolge von Nachrichten, die zwischen den Objekten ausgetauscht werden, und gibt einen Überblick darüber, wie das System funktioniert. Ein einfaches Beispiel für ein Sequenzdiagramm könnte ein Bankautomatensystem sein. In diesem System gibt es verschiedene Objekte, wie zum Beispiel den Bankautomaten selbst, den Kunden und das Bankensystem.



Das Sequenzdiagramm könnte wie folgt aussehen:

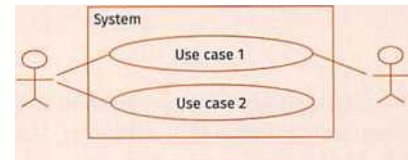


In diesem Beispiel gibt es drei beteiligte Objekte: den "Kunden", den "Bankautomaten" und das "Bankensystem". Der Kunde wählt zunächst die Auszahlungsoption und gibt den gewünschten Betrag ein. Der Bankautomat sendet dann eine Anfrage an das Bankensystem, um das Konto und die PIN des Kunden zu überprüfen. Wenn die Überprüfung erfolgreich ist, gibt der Kunde den Betrag ein und der Bankautomat prüft, ob der Betrag verfügbar ist. Wenn der Betrag verfügbar ist, gibt der Bankautomat das Geld aus und aktualisiert den Kontostand des Kunden im Bankensystem. Das Sequenzdiagramm gibt einen Überblick über die

Interaktionen zwischen den Objekten und zeigt, welche Nachrichten in welcher Reihenfolge ausgetauscht werden. Es ist ein wichtiges Werkzeug bei der Planung und Entwicklung von Systemen, die verschiedene Objekte enthalten und miteinander interagieren.

Anwendungsfalldiagramm (Use Case Diagram)

Das Use Case Diagramm ist eine Diagrammart in der UML, die verwendet wird, um die



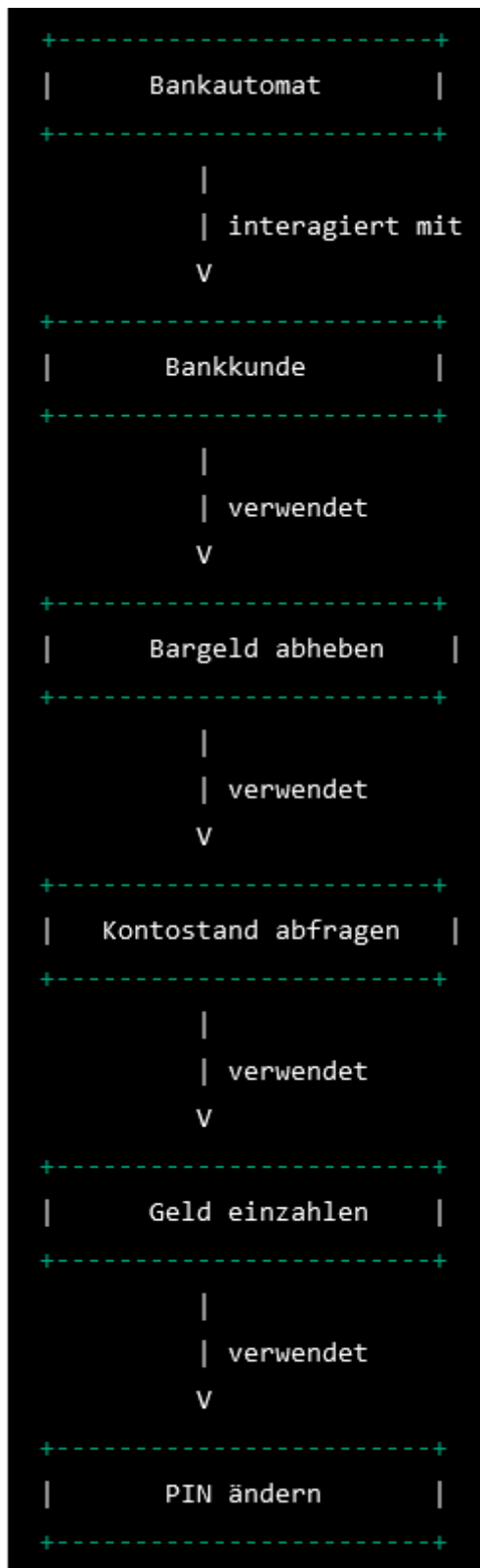
Interaktionen zwischen dem System und seinen Akteuren (Benutzern, Geräten oder anderen Systemen) darzustellen. Es zeigt die Hauptanwendungsfälle (Use Cases) des Systems und die Beziehungen zwischen ihnen und den Akteuren.

Ein Beispiel für ein Use Case Diagramm könnte ein Bankautomatensystem sein. In diesem System gibt es zwei Hauptakteure: den Bankkunden und den Bankautomaten. Die Hauptanwendungsfälle des Systems könnten folgende sein:

- Bargeld abheben
- Kontostand abfragen
- Geld einzahlen
- PIN ändern

Das Use Case Diagramm könnte wie folgt aussehen:

In diesem Beispiel zeigt das Use Case Diagramm, dass der Bankkunde über den Bankautomaten auf das Banksystem zugreift, um Bargeld abzuheben, den Kontostand abzufragen, Geld einzuzahlen oder die PIN zu ändern. Der Bankautomat fungiert dabei als Vermittler zwischen dem Bankkunden und dem Banksystem. Durch das Use Case Diagramm kann man schnell einen Überblick über die



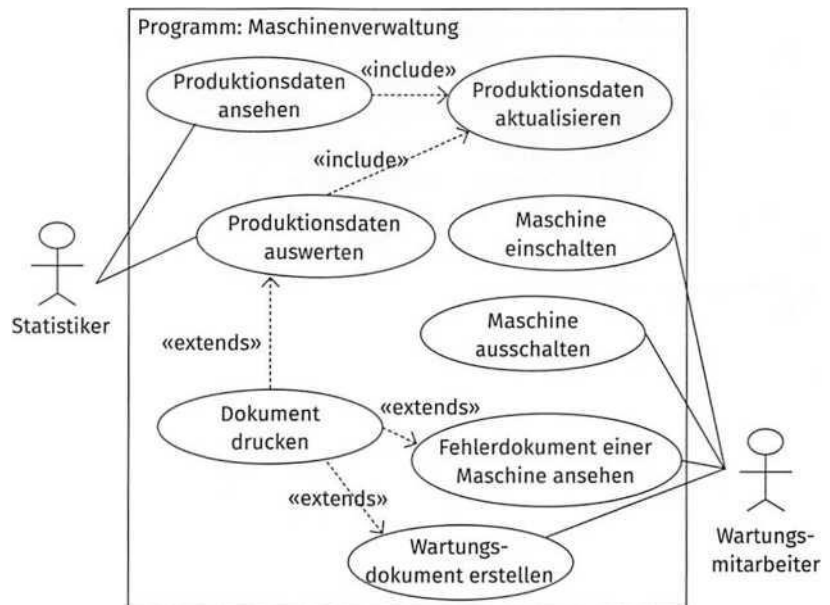
Hauptfunktionalitäten des Systems und die Beziehungen zwischen den Akteuren und den Use Cases erhalten.

Umsetzung eines Anwendungsfalldiagramms für die Ausgangssituation

Für die Ausgangssituation könnte eine Planung für die Anwendungsfälle wie in der Abbildung des Anwendungsfalldiagramms dargestellt aussehen.

Dieses Anwendungsfalldiagramm kann folgendermaßen gelesen werden: Die Wartungsmitarbeiter der Firma sollen mithilfe des Programms Maschinen vom Maschinenpark ein- und ausschalten können. Des Weiteren sollen sie in der Lage sein, sich das Fehlerprotokoll einer Maschine anzusehen oder ein Wartungsprotokoll für diese zu erstellen. Beide

Dokumente sollen bei Bedarf ausgedruckt werden können. Die Statistiker der Firma sollen sich mithilfe des Programms die Produktionsdaten der Maschinen ansehen und statistisch auswerten können. Dazu müssen die Produktionsdaten vom Programm jedes Mal aktualisiert werden. Auch für die Statistiken soll die Möglichkeit des Ausdrucks bestehen. Der Prozess der Wartung der Maschinen und der Auswertung der Produktionsdaten wurde hier sehr vereinfacht dargestellt. Für eine komplexere Aufgabenstellung würde auch das Anwendungsfalldiagramm komplexer ausfallen. Alternativ können auch mehrere Anwendungsfalldiagramme dann erstellt werden.



Weitere Beispielaufgabe zum Anwendungsfalldiagramm

Beispielaufgabe

Im Rahmen eines Kundenauftrages soll ein Programm für eine Onlinereservierung von Hotelzimmern für eine Hotelkette entwickelt werden. In einem ersten Gespräch sind schon einige Informationen über die Funktionalität des Programms erörtert worden. So sollen die Kunden u.a. Zimmer reservieren können. Der Reservierungsvorgang schließt immer mit ein, dass das Programm nochmal prüft, ob das entsprechende Zimmer zur Verfügung steht. Die Reservierung schließt weiterhin die Angabe einer Zahlungsmethode mit ein. Wird die Zahlungsmethode „Kreditkartenzahlung“ gewählt, dann soll das Programm die Kreditkartenangaben bei der entsprechenden Bank überprüfen. Dabei stellt die Bank ein anderes externes System dar, welche mit der Anwendung interagiert. Sie sollen aus diesen Angaben ein Anwendungsfalldiagramm entwickeln, welches als Diskussionsgrundlage für weitere Treffen mit dem Auftraggeber angedacht ist.

Lösung

Zur Lösung wird zunächst einmal das System als Rechteck gezeichnet und ein entsprechender Name vergeben. Danach werden die Akteure und die Anwendungsfälle laut Aufgabenstellung analysiert und in das Diagramm eingezeichnet. In diesem Fall sind die Akteure die Kunden, welche das Programm benutzen und die



Bank als ein anderes externes System, welches mit dem Programm interagiert. Zu den Anwendungsfällen zählen „Zimmer reservieren“, „Verfügbarkeit prüfen“, „Zahlungsmethode eingeben“ und „Kreditkarte prüfen“. Danach werden die Akteure und die Anwendungsfälle in Beziehung gesetzt. Das entsprechende Anwendungsfalldiagramm sieht dann wie folgt aus.